IN THE CLAIMS:

1. (Currently Amended) For use in a processor having an at least four-wide instruction issue architecture and instruction grouping logic, a mechanism for pipeline processing multiply-accumulate instructions with out-of-order completion, comprising:

a multiply-accumulate unit (MAC) having an initial multiply stage and a subsequent accumulate stage; and

out-of-order completion logic, associated with said MAC, that causes interim results produced by said multiply stage to be stored when said accumulate stage is unavailable and allows younger instructions to complete before said multiply-accumulate instructions, said grouping logic grouping said multiply-accumulate instructions based on said mechanism.

2. (Original) The mechanism as recited in Claim 1 wherein said initial multiply stage and said subsequent accumulate stage are single clock cycle stages.

3. (Original) The mechanism as recited in Claim 1 wherein said out-of-order completion logic is contained in a writeback stage of a pipeline in said processor.

4. (Original) The mechanism as recited in Claim 1 wherein said out-of-order completion logic writes back said interim results to at least one register in said MAC before said multiply-accumulate instructions arrive at said accumulation stage of said MAC.

5. (Original) The mechanism as recited in Claim 1 wherein said interim results are unavailable to an external program executing in said processor.

6. (Canceled)

7. (Original) The mechanism as recited in Claim 1 wherein said processor is a digital signal processor.

8. (Currently Amended) For use in a processor having an at least four-wide instruction issue architecture and instruction grouping logic, a method of pipeline processing multiply-accumulate instructions with out-of-order completion, comprising:

providing a multiply-accumulate unit (MAC) having an initial multiply stage and a subsequent accumulate stage;

causing interim results produced by said multiply stage to be stored when said accumulate stage is unavailable; and

allowing younger instructions to complete before said multiply-accumulate instructions, said grouping logic grouping said multiply-accumulate instructions based on said causing and said allowing.

9. (Original) The method as recited in Claim 8 wherein said initial multiply stage and said subsequent accumulate stage are single clock cycle stages.

10.(Original) The method as recited in Claim 8 wherein said causing is carried out in a writeback stage of a pipeline in said processor.

11. (Original) The method as recited in Claim 8 wherein said causing comprises writing back said interim results to at least one register in said MAC before said multiply-accumulate instructions arrive at said accumulation stage of said MAC.

12. (Original) The method as recited in Claim 8 wherein said interim results are unavailable to an external program executing in said processor.

13. (Canceled)

14. (Original) The method as recited in Claim 8 wherein said processor is a digital signal processor.

15. (Currently Amended) A digital signal processor (DSP), comprising:

a pipeline having stages and capable of processing multiply-accumulate instructions;

an instruction issue unit, containing grouping logic and at least four-wide instruction issue logic;

a multiply-accumulate unit (MAC), coupled to said instruction issue logic, having an initial multiply stage and a subsequent accumulate stage; and

out-of-order completion logic, associated with said pipeline, that causes interim results produced by said multiply stage to be stored when said accumulate stage is unavailable and allows younger instructions to complete before said multiply-accumulate instructions~~,~~, said grouping logic grouping said multiply-accumulate instructions to execute in a single clock cycle.

16. (Original) The DSP as recited in Claim 15 wherein said initial multiply stage and said subsequent accumulate stage are single clock cycle stages.

17. (Original) The DSP as recited in Claim 15 wherein said out-of-order completion logic is contained in a writeback stage of said pipeline.

18. (Original) The DSP as recited in Claim 15 wherein said out-of-order completion logic writes back said interim results to at least one register in said MAC before said multiply-accumulate instructions arrive at said accumulation stage of said MAC.

19. (Original) The DSP as recited in Claim 15 wherein said interim results are unavailable to an external program executing in said DSP.

20. (Canceled)